

Assembling an Enterprise Application Integration Toolkit

a report by

Douglas W Allen

Consulting Information Technology Architect,

US Enterprise Application Integration (EAI) National Practice, IBM Global Services

Enterprise application integration (EAI) has rapidly evolved from a buzz word into a multibillion-dollar software market segment. Predictably, this space is flooded with products and vendors, each claiming to offer everything customers need to integrate their entire enterprise. Brochures, flyers and websites that promise end-to-end solutions for business-to-business, business-to-consumer, EAI, etc., have made the task of choosing a product – or a combination of products – to meet an enterprise's integration needs, daunting.

This article presents procedures for gathering requirements, conducting research and selecting a set of EAI tools best suited for integrating a unique enterprise. It offers advice on the specific information that should be obtained from vendors when evaluating integration solutions to determine functionality, as well as the impact that integration tools might have on an existing information technology (IT) environment.

Assess the Current Information Technology (IT) Environment

This crucial step is often overlooked when EAI tools are brought into an organisation. Tools are often introduced, only to find that they are not well suited to the applications that require immediate integration. Such problems can be avoided, by first gaining a simple understanding of the applications that require integration. Examine the existing applications and decide which are strategic (i.e. those that support the organisation's strategic business initiatives), conduct interviews with leading IT personnel and determine their integration needs.

When evaluating the strategic applications from an EAI perspective, integration points need to be identified. The architecture of the applications also needs to be examined in order to determine whether they have been deployed using some sort of distributed object or component model, or whether they are large, monolithic, 'stovepipe' applications. Those implemented via standard component models are more easily 'integratable' because of their greater number of defined interfaces.

The existing infrastructure should be examined, and database technologies, transaction processing monitors and other enabling technologies that are currently being used in the organisation should be determined, as it is likely that these products will be participating in some of the integration scenarios.

Determination of Integration Tools

Like any other buzz word, the term EAI has come to mean many different things to many different people. Before looking for specific products, there should first be some understanding of the EAI marketplace as a whole. Generally, it can be separated into the following primary categories of software services, bearing in mind that there is a great deal of overlap in the services provided by products in each of these categories:

- Asynchronous event/message transport – these products enable asynchronous routing of business events between applications. This service is provided typically by message-oriented middleware (MOM) products, but can be accomplished by other means as well. Including this service is key, due to the unpredictable nature of many integration scenarios. It is likely that a variety of applications that have a variety of availability windows will need to be integrated. Therefore, the ability to defer delivery of business events until applications are available will be needed. This service also ensures that applications can maintain loosely coupled relationships, which is a fundamental EAI solution design principle.
- Transformation engines – tools providing transformation services feature the ability to convert data and business events from one format to another. For example, a transformation engine could be used to convert customer records from an enterprise resource planning (ERP) system into formats required by a heritage, 'home-grown' customer service application. Transformation engines are typically batch-oriented and operate at the database table or file level.
- Integration brokers – also known as message brokers, these tools provide the ability to



Douglas W Allen is a consulting information technology (IT) architect, currently responsible for the development of the Business Process Management Competency Segment within IBM Global Services' US Enterprise Application Integration (EAI) National Practice. With more than 11 years of experience in the IT field, Mr Allen has focused on researching, designing and deploying distributed middleware and EAI-oriented applications since 1995. He has designed and implemented solutions using a variety of EAI tools and programming languages across a variety of operating system platforms. He has published numerous articles on the EAI discipline and is currently developing EAI-oriented curricula and training materials for IBM Global Services.

intelligently route and manipulate business events between multiple applications and data stores. For example, an integration broker could receive order requests from a Web-based application and route those requests to one or more target applications based on information in the order. The source event would be transformed into the format expected by the destination applications. Therefore, by definition, integration brokers also include data transformation services. Integration brokers support an event paradigm and deal with individual records, rows or autonomous business events.

- Business process management frameworks – these tools, which enable business logic to be separated from process-flow logic, consist of two distinct categories – process automation and work flow.

Process automation products provide a framework that allows multiple disparate software components to participate in an integrated business process flow. By this definition, a software component could be anything from a single-purpose module, such as a calculation function within a shared library, to an update of order information in an ERP system. For organisations that have embraced the concepts of object-oriented programming, process automation frameworks allow maximum reuse of application logic across an enterprise.

Work flow products support a similar paradigm, but focus on process steps performed by human interactions with the system. As business events move through a process flow, they will require routing and transformation services. Therefore, by definition, process automation/work flow frameworks also include either their own integration broker services or the ability to use the services of an external integration broker.

Products in all of these classes of software also include a variety of adapters that provide connectivity to leading applications, databases and middleware. Some provide frameworks that ease the construction of adapters to home-grown, custom and/or non-supported applications and data stores.

Understanding these services enables categorisation of the various EAI software vendors. Additionally, understanding the various categories of EAI services will help refine the EAI requirements specific to the enterprise, which will prevent unnecessary, superfluous purchases.

Product Comparison Testing

Once EAI services requirements have been examined, and a few vendors whose products provide those services have been settled on, there

needs to be some comparison testing conducted. A simple integration scenario needs to be visualised that represents an EAI problem in the enterprise. Each vendor should be brought in for a set amount of time to work with the team and develop a prototype solution for the integration scenario.

Most vendors have pre-sales or services organisations that will aid in these efforts. As the EAI marketplace is relatively new, this product comparison testing is an absolute necessity. Marketing literature discussing the vendor's visions, customer success stories, industry awards and stock performance are no substitute for a team actually utilising each set of tools and products for testing purposes.

During the testing, it is important to focus on the overall architecture of the product and examine the relationships between each component. If the product is a complete process automation framework, an understanding of the relationships between the process engine, the transformation components and the adapters should be gained.

There are many questions that need to be asked in order to provide an idea as to the openness of the entire tool set and some insight into the future direction of the product.

The first of these is whether the vendor takes advantage of third-party databases for the persistence of rules, routing information and business event formats, or whether it has its own proprietary file structure. If the vendor allows the use of third-party databases, it should be ascertained which are currently supported and which it plans to support in the future, and whether the vendor is planning to shift from proprietary to third-party, or vice versa. It should also be determined whether the overall framework allows easy advantage of services offered by other vendors' products.

When evaluating business process management frameworks, it needs to be ascertained whether the process modeller conforms to a standard, such as Unified Modeling Language (UML), or whether the vendor uses its own set of symbols, in addition to whether the tool can export and import process models to and from other modelling tools. It should be determined whether the process modeller generates the 'under-the-covers' code needed to execute the business process, or if it is simply a drawing tool. If it generates the code, it needs to be determined whether it is in a common language, such as Java, or the vendor's own proprietary language. Also to be checked is how the code is stored, compiled and executed.

How changes to the process models are propagated through to the actual execution of the underlying code needs to be determined, and whether it can be

Figure 1: Simple Order Processing System



performed dynamically, or whether the entire environment must be shut down and restarted for the changes to take effect. If a step in a given process fails, it should be ascertained how other steps in the process are backed out, if at all.

It should also be determined whether a given process can be monitored for both debugging and business purposes. For debugging purposes, developers and support personnel will likely need to detect which step in the process is stalled. From a business perspective, systems should be developed that enable end-users to monitor the status of steps in a process. For example, customer service representatives should be able to query the status of customers' orders, payments, and so on.

With business process management frameworks and integration brokers, a solid understanding should be gained of how the product defines an 'event',

sometimes referred to by vendors as 'business objects' or 'message formats'. Events are merely representations of autonomous pieces of data that flow through the integration environment. Ideally, they represent autonomous business events that were generated by changes in the applications participating in an integration scenario.

As for event formats, many integration brokers provide the ability to import and export XML document type definitions (DTDs), COBOL copybooks and C-language header files. These features should be comparison tested, as they can eliminate a great deal of tedious work by development staff. Some tools can also query existing database tables and generate representative event formats based on the result.

It should also be determined whether the product provides true publish-subscribe ("pub/sub") capability,

Perspective Software Limited

Enterprise Application Integration Consultants

From this:

To this:

Why should you care about EAI ? Forrester Research & Gartner estimate that developers spend up to 40% of their time creating interfaces and points of integration for applications and data sources outside of their control.

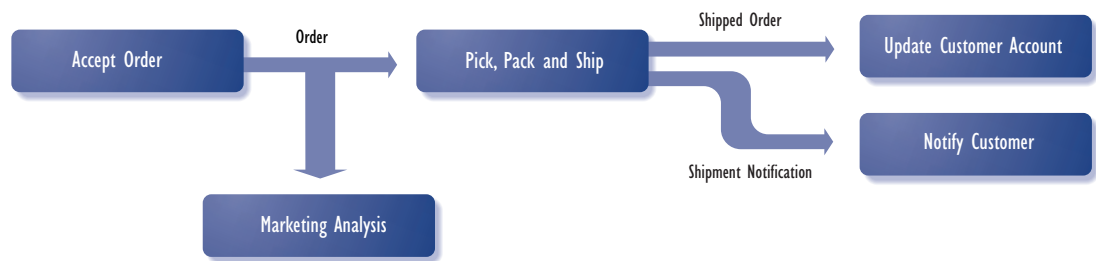
Integration, not perspiration EAI provides a set of tools to integrate all your systems into a coherent enterprise architecture, regardless of platform or programming language whether in-house or packaged; legacy, client-server or web based.

Perspective Software specialises in providing EAI solutions to the financial services industry. To kick start your integration efforts, we offer an EAI assessment, providing you with a roadmap for your future integration projects.

E-Mail info@psw.ie or tel 353 1 6384890

Business partners of

Figure 2: Introduction of Additional Subscriber-to-order Events



as vendors often use these terms with multiple meanings. Essentially, it needs to be ascertained whether the products allow additional entities (applications, components, processes, whatever is being integrated) to register their interest in a given event type without disrupting the operation or configuration of existing entities that are currently using that event. For example, it may be decided upon to use a process automation framework to build a solution for processing customer orders, as illustrated in the unrealistic and oversimplified model in *Figure 1*, which shows an order event flowing from the order entry system to the warehouse system and to the accounting system. Upon shipment, a shipment notification event is also sent to a system that will notify the customer that the order is on its way. The marketing department may also be interested in incoming order events so that its systems can perform some sort of analysis of sales trends, as illustrated in *Figure 2*.

A true pub/sub framework would allow the marketing analysis application to also receive the incoming order events without disrupting or modifying anything in the operation or configuration of the existing order processing flow. If and when the sales trend analysis is complete, the marketing analysis application should be able to ‘unsubscribe’ to the event with similar minimal impact.

When evaluating integration brokers and transformation engines, tools that mix ‘drag-and-drop’ functionality with programming or scripting to define rules for transforming one event type into another will be looked at. Finding out if the tool provides a set of function libraries that allow operations beyond simple field-to-field data mapping is suggested, as functions such as lower-case/upper-case conversion, concatenation, substrings, etc., may be available.

Additionally, integration brokers and transformation engines should allow the performance of ‘event augmentation’ through external database queries to aid in conversions. This feature will allow the definition of some of the transformation rules in an external database, which will add a dynamic element to the operation of the transformation engine. In other words, the behaviour of the transformation engine changes as the values in the database change.

It should also be determined if the tool has the ability to perform some sort of external procedure call. This will allow the reuse of any existing transformation-related functions there may be in classes or libraries written in languages such as Java or C.

The adapters that provide connectivity from the integration tools to the participants in the integration scenarios – software components and data stores – should be scrutinised. If the vendor provides a database connector, it should be determined whether it provides ‘native’ connectivity or uses open database connectivity (ODBC) or Java database connectivity (JDBC). This will determine how the system will be configured, and there may be performance implications.

In order to receive incoming events from changes to application database tables, some products require the creation of additional tables – event or shadow tables – in the database. This technique requires the addition of triggered stored procedures that populate the event/shadow tables based on changes to the application tables being monitored. The adapter then submits events to the integration environment by polling the event/shadow tables, usually at some configurable interval.

Other database adapters allow repeated execution of a specific Structured Query Language (SQL) query to obtain events. In any case, the administrators of the application databases should be consulted during the comparison testing to enable them to ask questions and voice their concerns about a particular vendor’s connectivity approach, since some approaches may have a negative effect on database performance.

If the vendor provides connectivity to the MOM of choice, ensure that it is determined how that connectivity is achieved and whether the adapter enables exploitation of any of the functionality provided by the MOM, such as the selection of messages based on message properties or header information, should be checked.

If the vendor provides an adapter to a commercial off-the-shelf (COTS) system, details should be obtained about connectivity. It should be determined whether the connectors provide realtime connectivity at the

application or data level, or whether they merely generate files that must then be imported into the COTS environment through some other means. Most of the major COTS vendors are constantly developing and enhancing access methods to ease integration with their systems and the onus is on the user to ascertain which methods the vendor uses. This information will allow the determination of whether a particular version of the COTS package provides the interface supported by the EAI tool vendor. The user should also be able to judge the integration vendor's commitment to supporting the latest release of the COTS package.

As mentioned, the administrators and owners of the home-grown applications and COTS packages should be involved in adapter discussions with the EAI vendor. Not only is it beneficial politically, but there may be many technical issues to be resolved. Obviously, it is better to have these issues settled when the products are being evaluated, rather than implemented.

The Enterprise Application Integration (EAI) Services Platform

Selecting the appropriate tools is only the first step in constructing an EAI services platform. Teams also need to concern themselves with such tasks as:

- contract negotiation with the EAI vendor(s);
- training employees to use the EAI tools;
- system installation and configuration;
- establishing management and support procedures for the EAI services platform; and
- selecting the applications that will be the first to use the EAI services platform.

It is easy to become caught up in the hype surrounding the EAI marketplace and its promise of a fully integrated enterprise. Although marketing materials and vendor demonstrations serve a purpose, they merely scratch the surface when describing the functionality and features of any EAI tool.

A true evaluation can only be performed by sitting down and using the tools in each individual environment. Gathering specific information about the issues presented in this article will allow for a more informed decision to be made and will alleviate many problems that arise during the implementation phases of EAI projects. ■

Copyright © 2001 by ebizQ (<http://www.ebizQ.net/>)

Marketing your products & services? Raising your corporate profile? Attending an exhibition?

Our in-house design team can create a presentation package specifically tailored to suit your company's marketing requirements, which you can then use for exhibitions, conferences and other promotional purposes.

We offer special rates to advertisers for the reproduction of reprints from 10 to 10,000 copies of your advertisement, corporate presentation and/or technical article.



For further details and a quote for this service, please contact:
Tina Price, Publisher, World Markets Research Centre
Cardinal Tower, 12 Farringdon Road, London EC1M 3NN
Tel.: +44 (0)20 7452 5034, Fax: +44 (0)20 7452 5050, E-mail: tina.price@wmrc.com



**World Markets
Research Centre**